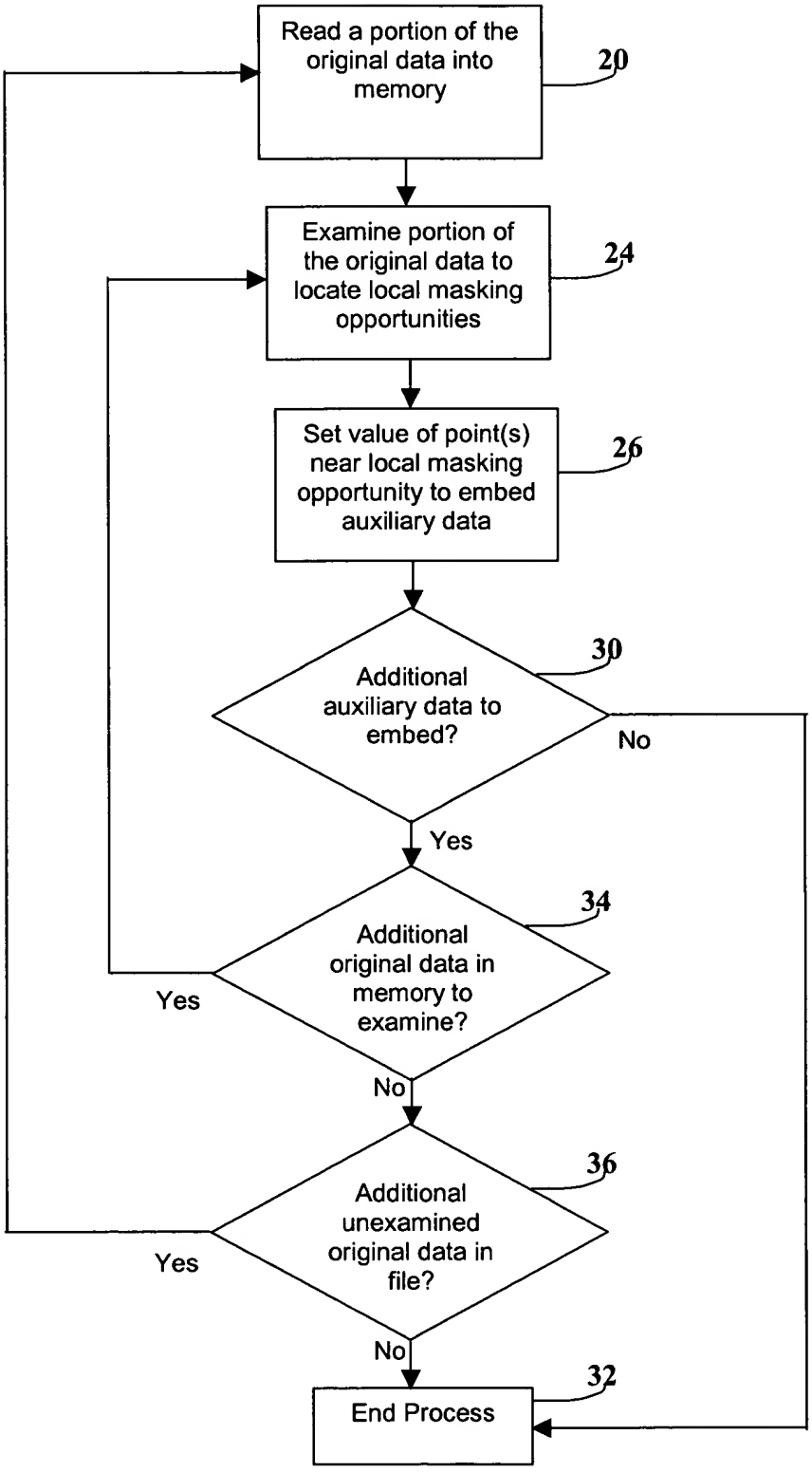
Read a portion of the original data into memory
20
Examine portion of the original data to locate local masking opportunities
24
Set value of point(s) near local masking opportunity to embed auxiliary data
26
Additional auxiliary data to embed?
30
No
Yes
Additional original data in memory to examine?
34
Yes
No
Additional unexamined original data in file?
36
Yes
No
End Process
32

Fig 1

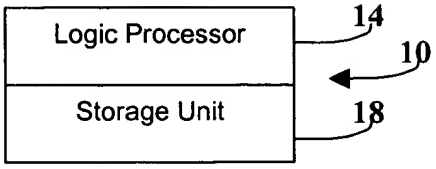

Fig. 2

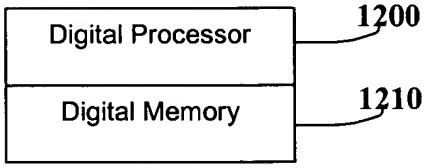

Fig. 12

A to D Converter	Sample and Hold	D to A Converter
Comparator	PLC	Delays

Fig 13

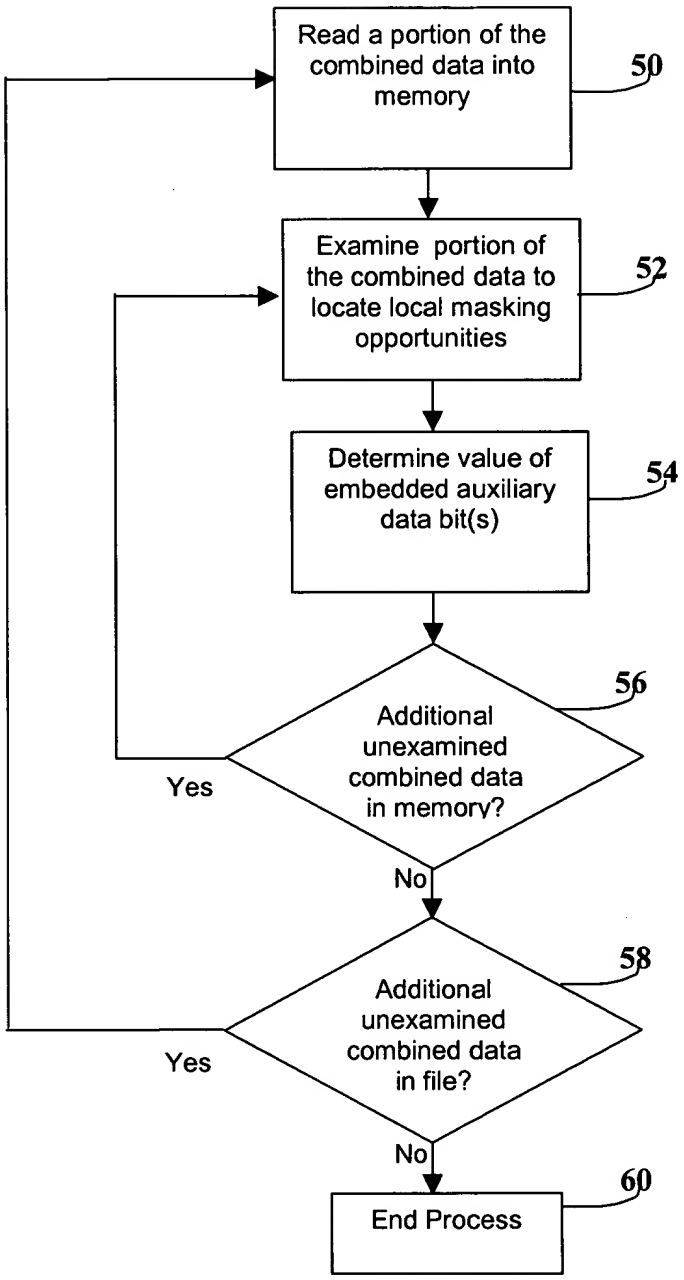
Read a portion of the combined data into memory
50
Examine portion of the combined data to locate local masking opportunities
52
Determine value of embedded auxiliary data bit(s)
54
Additional unexamined combined data in memory?
56
Yes
No
Additional unexamined combined data in file?
58
Yes
No
End Process
60

Fig 3

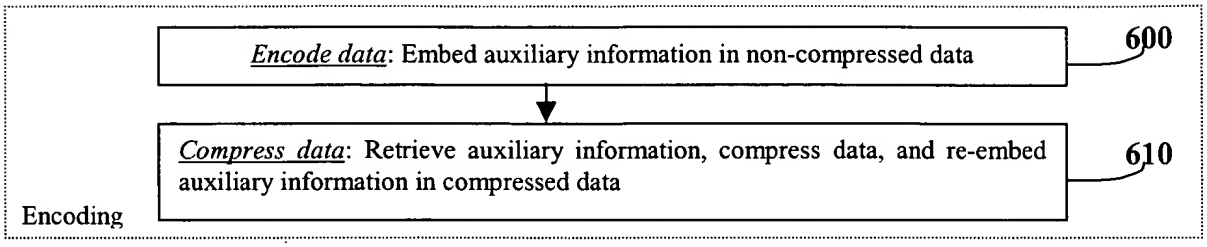

Fig. 10A

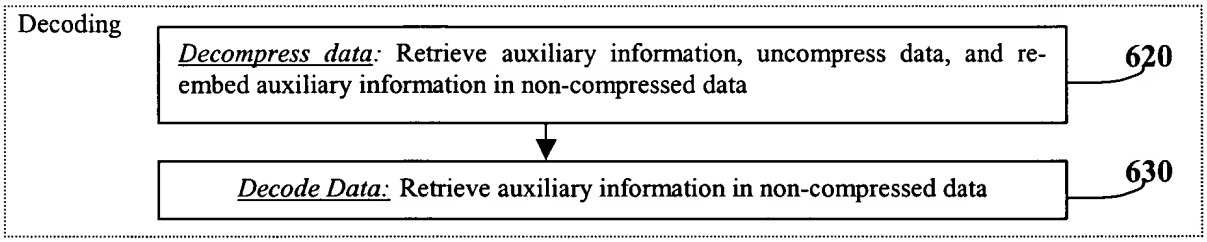

Fig. 10B

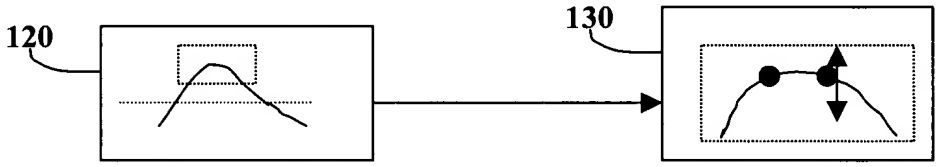

Fig. 4

Fig. 7

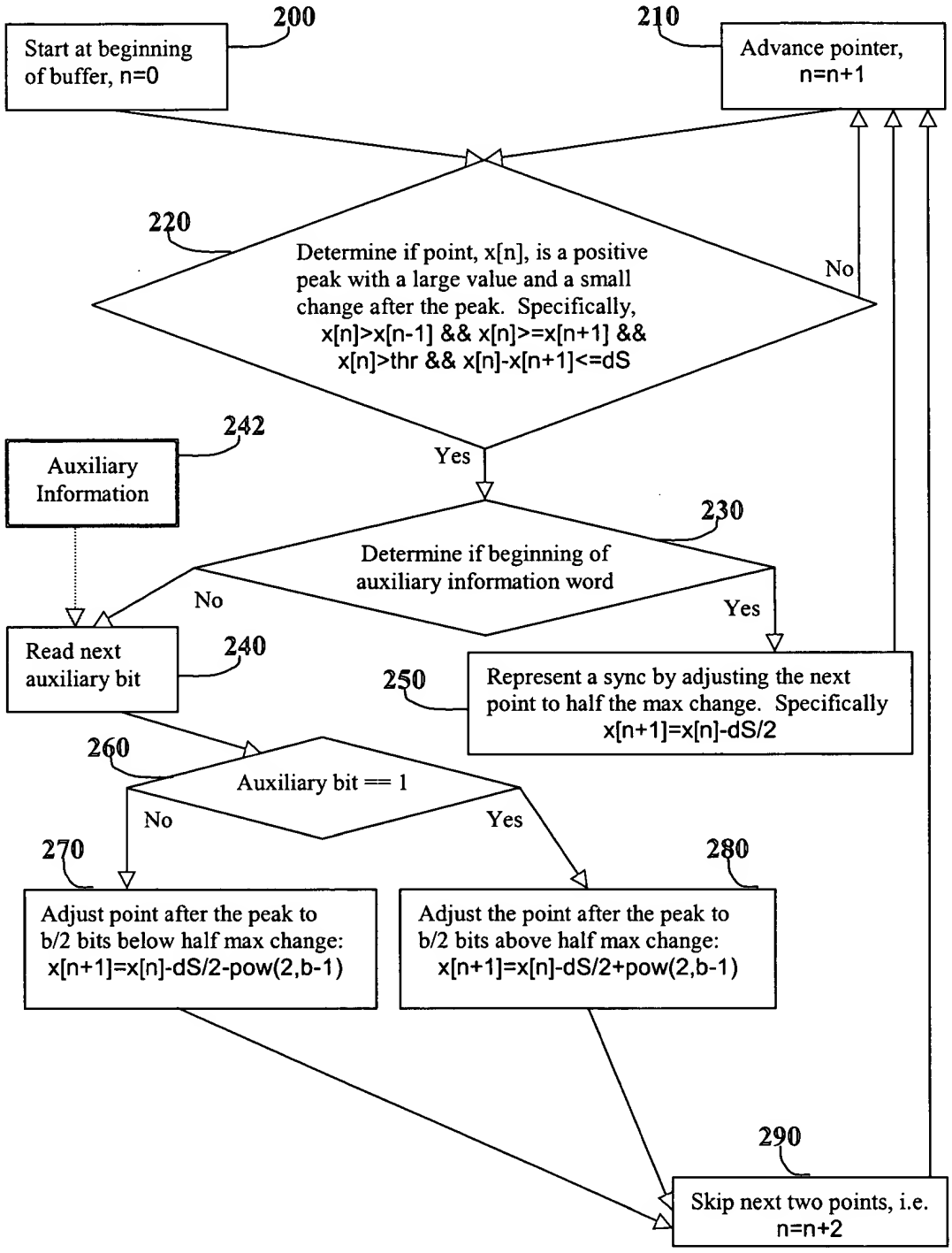
200
Start at beginning of buffer, n=0
210
Advance pointer, n=n+1
220
Determine if point, x[n], is a positive peak with a large value and a small change after the peak. Specifically, x[n]>x[n-1] && x[n]>=x[n+1] && x[n]>thr && x[n]-x[n+1]<=dS
No
Yes
242
Auxiliary Information
230
Determine if beginning of auxiliary information word
No
Yes
Read next auxiliary bit
240
250
Represent a sync by adjusting the next point to half the max change. Specifically x[n+1]=x[n]-dS/2
260
Auxiliary bit == 1
No
Yes
270
Adjust point after the peak to b/2 bits below half max change: x[n+1]=x[n]-dS/2-pow(2,b-1)
280
Adjust the point after the peak to b/2 bits above half max change: x[n+1]=x[n]-dS/2+pow(2,b-1)
290
Skip next two points, i.e. n=n+2

Fig. 5

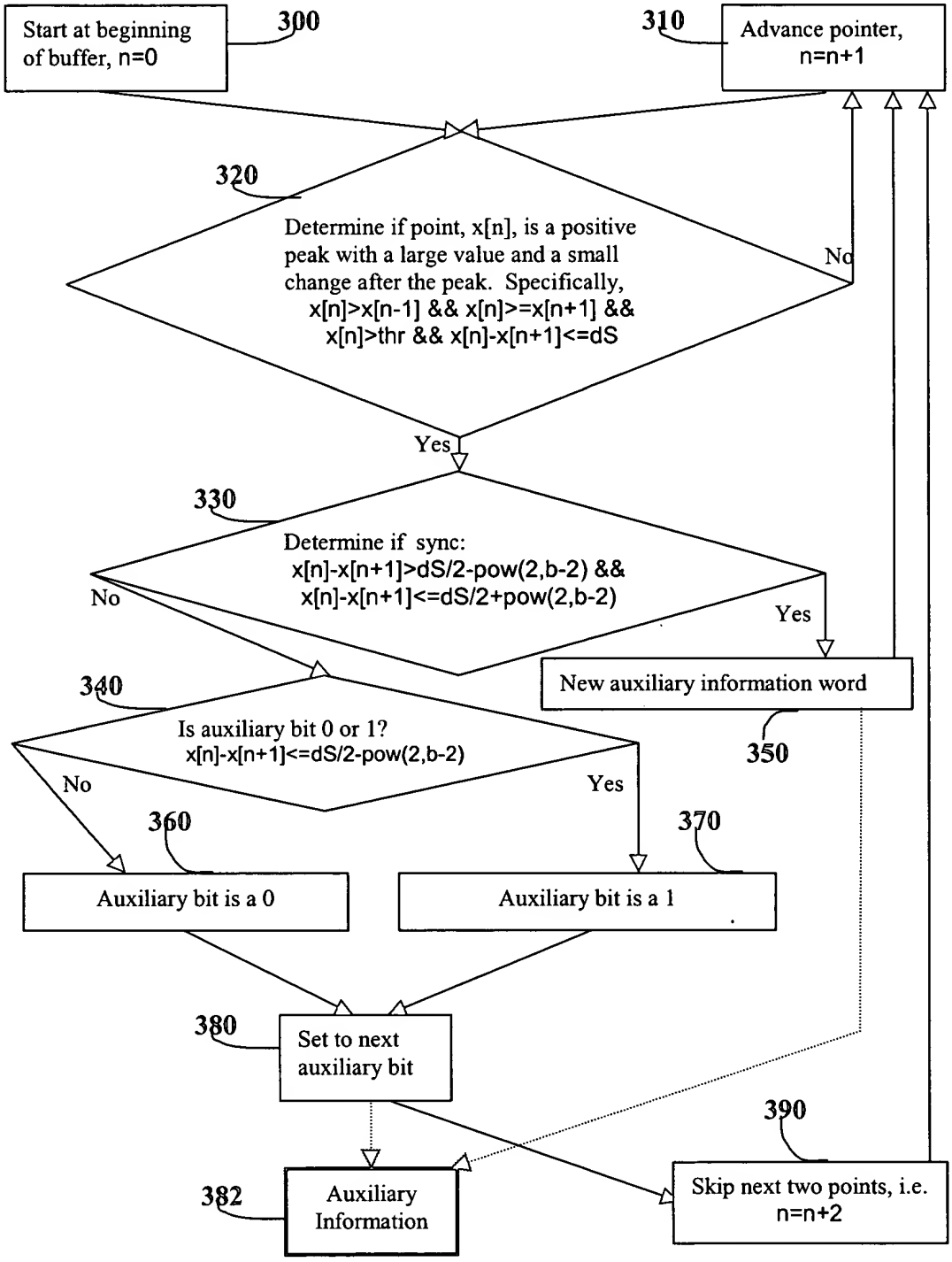
Start at beginning of buffer, n=0
300
310
Advance pointer, n=n+1
320
Determine if point, x[n], is a positive peak with a large value and a small change after the peak. Specifically, x[n]>x[n-1] && x[n]>=x[n+1] && x[n]>thr && x[n]-x[n+1]<=dS
No
Yes
330
Determine if sync: x[n]-x[n+1]>dS/2-pow(2,b-2) && x[n]-x[n+1]<=dS/2+pow(2,b-2)
No
Yes
New auxiliary information word
350
340
Is auxiliary bit 0 or 1? x[n]-x[n+1]<=dS/2-pow(2,b-2)
No
Yes
360
370
Auxiliary bit is a 0
Auxiliary bit is a 1
380
Set to next auxiliary bit
390
382
Auxiliary Information
Skip next two points, i.e. n=n+2

Fig. 6

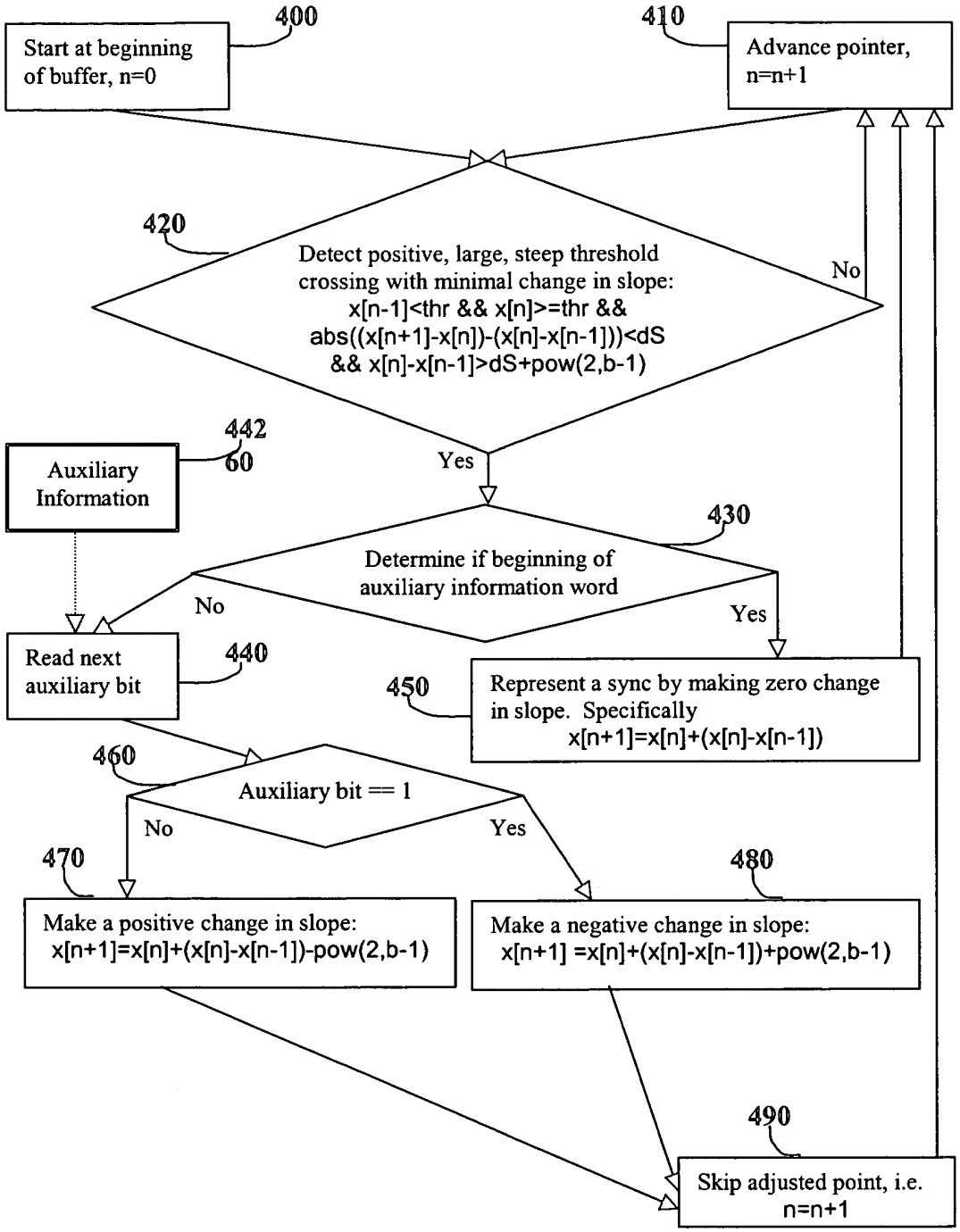
400
Start at beginning of buffer, n=0
410
Advance pointer, n=n+1
420
Detect positive, large, steep threshold crossing with minimal change in slope:
x[n-1]<thr && x[n]>=thr &&
abs((x[n+1]-x[n])-(x[n]-x[n-1]))<dS
&& x[n]-x[n-1]>dS+pow(2,b-1)
No
Yes
442
60
Auxiliary Information
430
Determine if beginning of auxiliary information word
No
Yes
Read next auxiliary bit
440
450
Represent a sync by making zero change in slope. Specifically
x[n+1]=x[n]+(x[n]-x[n-1])
460
Auxiliary bit == 1
No
Yes
470
Make a positive change in slope:
x[n+1]=x[n]+(x[n]-x[n-1])-pow(2,b-1)
480
Make a negative change in slope:
x[n+1] =x[n]+(x[n]-x[n-1])+pow(2,b-1)
490
Skip adjusted point, i.e. n=n+1

Fig. 8

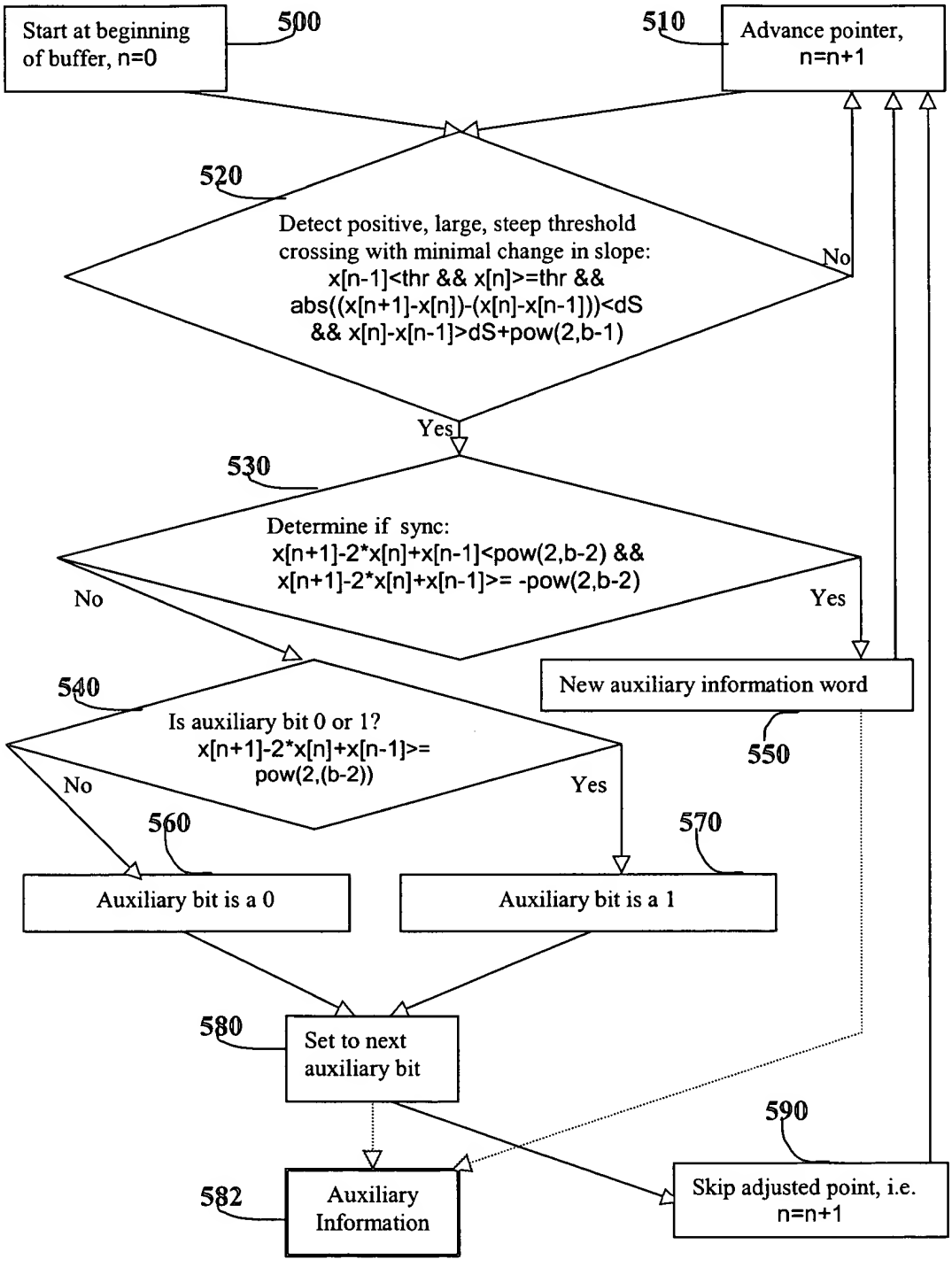
Start at beginning of buffer, n=0
500
510
Advance pointer, n=n+1
520
Detect positive, large, steep threshold crossing with minimal change in slope:
x[n-1]<thr && x[n]>=thr &&
abs((x[n+1]-x[n])-(x[n]-x[n-1]))<dS
&& x[n]-x[n-1]>dS+pow(2,b-1)
No
Yes
530
Determine if sync:
x[n+1]-2*x[n]+x[n-1]<pow(2,b-2) &&
x[n+1]-2*x[n]+x[n-1]>= -pow(2,b-2)
No
Yes
New auxiliary information word
550
540
Is auxiliary bit 0 or 1?
x[n+1]-2*x[n]+x[n-1]>=
pow(2,(b-2))
No
Yes
560
570
Auxiliary bit is a 0
Auxiliary bit is a 1
580
Set to next auxiliary bit
590
582
Auxiliary Information
Skip adjusted point, i.e. n=n+1

Fig. 9

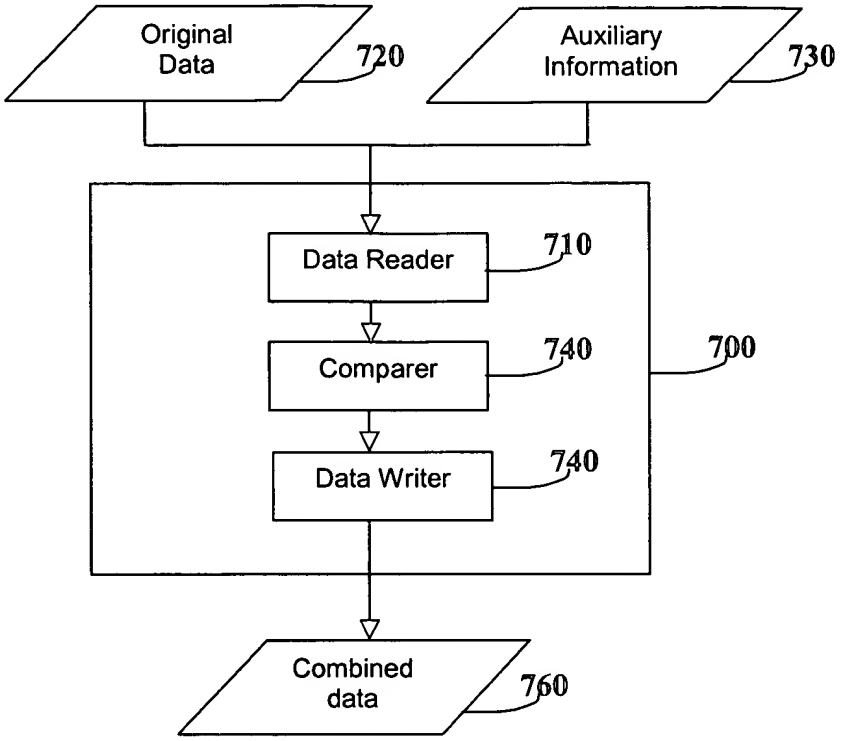

Fig. 11A

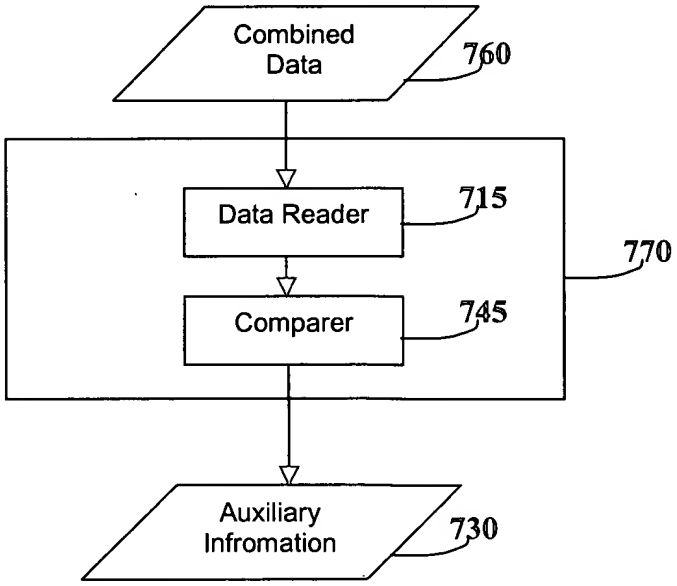

Fig. 11B